Serial No.: 09/781,928
Atty Docket No.: D5036

(b:) AMENDMENT TO SPECIFICATION

Please make the following corrections to the specification relating to errors occurring in the specification as published.

[0008] Software development can be greatly aided by increasing the level of abstraction at which the programmer works[,]. Increasing abstraction in a programming environment should be based on increasing independence from any one programming language or platform and allow sharing of fundamental components between programs. The flexibility and adaptability of such a system are enhanced by making the system extendible, which allows for evolution of the system and by providing a high level of robustness.

[0010] A programmer working in a COM environment is able to exploit a binary standard for function calling between components and to use common interfaces for strongly typed groups of functions. COM provides a base interface which allows components to dynamically discover the interfaces implemented by other components (through qualification) and which further provides reference counting to allow components to track their own lifetime and to delete themselves. See *The Component Object Model: A Technical Overview*, Williams, Sara & Kindel, Charles (1994).

[0018] **Fig. 1** is a perspective view of a vehicle ~~13~~ 11 and of an electrical control system **10** installed on the vehicle and vehicle chassis 13. Vehicle electrical system **10** comprises a network which may, in one embodiment, comprise a twisted pair (either shielded or unshielded) cable operating as a serial data bus **18**. One node of bus **18** is a body controller **30**, which is a major component of a vehicle electronic control system. Body controller **30** manages a number of vocational controllers connected to bus **18** as nodes. Collectively, bus **18** and the various nodes attached thereto form a controller area network (CAN). Numerous opportunities exist for programming the various controllers. Alternative

network classes may be employed.

[0019] Active vehicle components are typically controlled by one of a group of autonomous, vocational controllers, which include an instrument and switch bank **12**, a gauge cluster **14**, an engine controller **20**, a transmission controller **16**, and an antilock brake system (ABS) controller **22**~~, and a definable remote interface module 21,~~ all of which are connected to body controller **30** over a serial data bus **18** and all of which are connected to bus **18** as nodes. The autonomous controllers include local data processing and programming and are typically supplied by the manufacturer of the controlled component. For each autonomous controller there is a defined set of variables used for communications between the autonomous controller and other data processing components on the network or attached to the network. Bus **18** is preferably a twisted pair cable constructed in accordance with SAE standard J1939 and is externally accessible via a diagnostic port **36**. Diagnostic port **36** is typically located under the steering column inside the cab of vehicle **13**, but may be located elsewhere. Although the autonomous controllers handle many functions locally and are functionally ~~difficult~~ defined without reference to body controller **30**, they may report data to body controller **30** and may receive operational requests from body controller **30**. Gauge cluster **14**, transmission controller **16** and engine controller **20** all may communicate with body controller **30**, which also monitors inputs received from the auxiliary instrument and switch bank **12** over the serial communication link in harness **18**. For a CAN system a J1939 compliant cable **23** is connected from port **36** to a remote data link computer **44**.

[0021] **Fig. 2** is a block diagram of a data processing system **43** incorporating a data link PC **44**. Data link PC **44** is a personal computer of substantially conventional architecture, adapted to communicate with a vehicle network meeting either of the SAE protocols. It ~~has~~ is anticipated that any new protocols developed in the future will be as readily handled.

Computer **44** is attached to the usual peripheral devices including a display **60**, a keyboard **62** and a pointing device **64**. Pointing device **64** and keyboard **62** are connected to the computer's serial bus **65** which transmits inputs from those devices to the motherboard in a conventional manner. Computer **44** further includes a PCI bus for the attachment of adapter cards, which include a network adapter **66** and a PCI to CAN adapter card **76**. The network adapter **66** allows connection via an external ethernet **68** to other data processing equipment such as an event logger **70**. Adapter card **76** allows two connections to a vehicle CAN bus **92** over which communication may be ~~retained~~ maintained to a remote device **94**. Alternatively, remote device **94** may be connected by an SAE J1587 bus **96** to a port adapter **90** allowing communication between the remote device **94** and computer **44** through the computer's serial bus **65**. Such communication would occur over a conventional personal computer serial port. Motherboard **72** includes a special data bus **78** for communication with a floppy drive **80** a CD Rom **82** and a hard drive **84** and controls the display **60** over a local video bus **86** and display interface **88**. Remote data link personal computer **44** executes data link software package **100** described below to allow programmer interaction with remote device 94 attached to a motor vehicle network.

[0023] **Fig. 3** illustrates the component object model and interface design provided by a data link software package **100** of the invention. Software package **100** incorporates three COM interfaces through which a programmer may program or evaluate a vehicle network at an abstract level. Package **100** provides translation of a message moving between client and devices across a network boundary, relieving the programmer of concern for the details of specific network messaging. COM interfaces **38, 39** and **40** represent the highest level of abstraction of the system and, while believed by the present ~~invention~~ inventors to be a particularly convenient and logical way of presenting underlying systems, do not exhaust the possibilities for high level re-presentation. All COM interfaces must implement

Serial No.: 09/781,928
Atty Docket No.: D5036

the interface for a root COM component ~~á/Unknownñ~~ +/Unknown,. Every interface must have a unique interface identification (IID). An interface, once assigned an IID and published, becomes unmutable. Interface member functions should have a return type of HResult to allow reporting of remote procedure call (RPC) errors. String parameters in interface member functions should be in Unicode. Extensive descriptions of the Component Object Model are available from the Microsoft Online Library on the World Wide Web.

[0028] The function _Enumerate Networks_ **258** is used by clients to enumerate all physical networks, typically vehicle networks, that are currently available. A network is defined as available if an instance for the physical network has been created using the Acquire Network function **270[I]**. The client specifies a count of the number of INetwork pointers for which it is allocated storage. The data link software **100** then places as many enumerated networks into the network's array as possible. If there is insufficient storage, then the software returns only the count number of networks the client initially specified. Otherwise all networks that have been acquired are returned. When an acquired network is not called at all for an operating cycle, then count and networks return values are undefined and the HResult return code is set for fail. The return values for this function indicate success, a general failure that is a count or network's pointer, and a value for when no networks are successfully acquired. The variables include a count variable which is the number of networks that are in a set of networks returned to the client. Here the client specifies in a call the number of available network pointers in the network's variable. The client then receives the number of networks enumerated in this variable when the call returns. The second variable is the _networks_ variable, which is a pointer to an array of INetwork object pointers that describe the networks of which the data link software package **100** is currently aware.

Serial No.: 09/781,928
Atty Docket No.: D5036

[0033] An Enumerate Device function **426** is used to return the set of all devices that were detected the last time a detection occurred over the network. If there are no other remote devices on the network, then the set contains only the software's representation on the network. Variables for this operation include a Count variable which is the number of IRemoteDevice objects that are in the set of devices returned to the client. The client specifies on the call the number of available device pointers in the device less the number of variables. The client then receives the number of devices enumerated in this variable when the call returns. A device list variable is an array of IRemoteDevice objects that have been detected since the last network detection.

[0037] The _get_LastDeviceDetectionTime_ function ~~**682**~~ **681** returns the last time a device detection was performed on the underlying network. Its functional name is such that it can be used as a COM property or a function name. The sole variable for this function is the _last detection time_, a storage place for the last detection time of the network.

[0064] A _get_VehicleSystem_ function **567** returns the vehicle system code of this remote device on the underlying network. Its functional name is such that ~~is~~ it can be used as a COM property or a function name. The property is read only enforceable because there is no corresponding published put vehicle system that would be used for setting the property. One variable providing a storage place for the vehicle system represented by the object is defined for function **567**.